CLAIMS

1. A cache memory system, comprising:

a plurality of memory locations for storing data and addresses associated with the data, each of the plurality of memory locations having only a single word line associated therewith; and

at least one controller that enables first and second devices to access different ones of the plurality of memory locations concurrently.

2. The system of claim 1, wherein:

the memory locations are configured and arranged to be included in at least first and second ways; and

the at least one controller is configured and arranged to enable the first and second devices to concurrently access memory locations included in the first and second ways, respectively.

3. The system of claim 2, wherein the at least one controller is configured and arranged to give the first and second devices exclusive access to the first and second ways, respectively.

4. The system of claim 1, in combination with the first and second devices, wherein the first device includes a processor configured and arranged to access the memory locations, and wherein the second device includes a data transfer engine configured and arranged to transfer data between the memory locations and a lower-level memory.

5. The combination of claim 4, wherein the data transfer engine comprises a DMA controller.

6. A cache memory system, comprising:

a plurality of memory locations to store data and addresses associated with the data;

a plurality of cache outputs for providing data retrieved from the memory locations; and

first and second multiplexers having multiplexer inputs coupled to at least some of the memory locations and multiplexer outputs coupled to the plurality of cache outputs so as to enable the first and second multiplexers to select data from different ones of the plurality of memory locations to be provided concurrently on respective ones of the plurality of cache outputs.

7. The system of claim 6, wherein:

some of the memory locations are configured and arranged to form a data array having at least two ways, with each of the at least two ways having a respective data array output for providing data retrieved therefrom; and

the multiplexer inputs of the first and second multiplexers are coupled to the data array outputs so as to enable the first and second multiplexers to concurrently select data from different ones of the at least two ways of the data array to be provided concurrently on respective ones of the plurality of cache outputs.

8. A cache memory system, comprising:

a data array for storing data;

a tag array for storing tags associated with the data stored in the data array;

a load buffer coupled to the tag and data arrays to load tags and data into the tag and data arrays; and

a first multiplexer having an output coupled to an address input of the load buffer, the first multiplexer receiving as inputs first and second addresses from respective first and second sources, and providing as its output a selected one of the first and second addresses.

9. The system of claim 8, in combination with the first and second sources, wherein the first source includes a processor configured and arranged to access the tag and data arrays, and wherein the second source includes a data transfer engine configured and arranged to transfer data between the data array and a lower-level memory.

10. The combination of claim 9, wherein the data transfer engine comprises a DMA controller.

11. The system of claim 8, further comprising:

a second multiplexer having a first input coupled to an address output of the load buffer and a second input coupled to a source of third addresses, the second multiplexer providing as its output a selected one of the address output of the load buffer and one of the third addresses.

12. The system of claim 11, in combination with a data transfer engine that comprises the source of the second and third addresses.

13. The combination of claim 12, wherein the data transfer engine comprises a DMA controller.

14. The system of claim 8, further comprising:

a copy-back buffer coupled to the tag and data arrays to receive tags and data therefrom so that the received data can be transferred from the data array to a lower-level memory; and

a second multiplexer having an output coupled to an address input of the copy-back buffer, the second multiplexer receiving as inputs third addresses from the tag array and fourth addresses from a third source, and providing as its output a selected one of the third and fourth addresses.

15. A cache memory system, comprising:

a data array for storing data;

a tag array for storing tags associated with the data stored in the data array;

a load buffer coupled to the tag and data arrays to load tags and data into the tag and data arrays; and

a multiplexer having a first input coupled to an address output of the load buffer to receive first addresses therefrom and a second input coupled a source of second addresses, the multiplexer providing as its output a selected one of the first and second addresses.

16. The system of claim 15, in combination with the source of second addresses, which comprises a data transfer engine.

17. The combination of claim 16, wherein the data transfer engine comprises a DMA controller.

18. The system of claim 15, in combination with a lower-level memory, wherein the output of the multiplexer is used to load at least one memory word from the lower-level memory into the data array.

19. A cache memory system, comprising:

a data array for storing data;

a tag array for storing tags associated with the data stored in the data array;

a copy-back buffer coupled to the tag and data arrays to receive tags and data therefrom so that the received data can be transferred from the data array to a lower-level memory; and

a multiplexer having an output coupled to an address input of the copy-back buffer, the multiplexer receiving as inputs first addresses from the tag array and second addresses from a source distinct from the tag array, and providing as its output a selected one of the first and second addresses.

20. The system of claim 19, in combination with the source of second addresses, which comprises a data transfer engine.

21. The combination of claim 20, wherein the data transfer engine comprises a DMA controller.

22. A cache memory system, comprising:

a data array for storing data;

a tag array for storing tags associated with the data stored in the data array; and

at least first and second decoders adapted to receive and decode at least first and second respective addresses, the first decoder identifying, in response to receiving first addresses, first locations in the tag array and first locations in the data array corresponding to the first locations in the tag array, and the second decoder identifying,

in response to receiving second addresses, second locations in the tag array and second locations in the data array corresponding to the second locations in the tag array.

23. The system of claim 22, wherein each of the at least two decoders is preceded by a multiplexer that provides its output to that decoder, each of the multiplexers receiving the first and second addresses as respective inputs, and providing as its output a selected one of the first and second addresses.

24. The system of claim 22, wherein:

the tag and data arrays each have at least first and second ways;

the first locations in the tag array and the first locations in the data array are included, respectively, in the first way of the tag array and the first way of the data array; and

the second locations in the tag array and the second locations in the data array are included, respectively, in the second way of the tag array and the second way of the data array.

25. A cache memory system, comprising:

a data array including a first plurality of memory locations for storing data;

a tag array including a second plurality of memory locations for storing tags associated with the data stored in the data array; and

at least one controller configured to place the system in at least first and second states, wherein, in the first state, a first device has exclusive access to a first subset of the first plurality of memory locations and a second device has access to a second subset of the first plurality of memory locations, and, in the second state, the second device has access to at least one memory location in the first subset of the first plurality of memory locations.

26. The system of claim 25, wherein the at least one controller is configured such that, when the system is in the second state, the second device has exclusive access to the first subset of the first plurality of memory locations.

27. The system of claim 25, wherein the at least one controller is configured such that, when the system is in the first state, the second device has exclusive access to the second subset of the first plurality of memory locations.

28. The system of claim 26, wherein the at least one controller is configured such that, when the system is in the first state, the second device has exclusive access to the second subset of the first plurality of memory locations.

29. The system of claim 25, wherein:

the tag and data arrays each comprise at least first and second ways;

the first subset of the first plurality of memory locations is included in the first way of the data array; and

the second subset of the first plurality of memory locations is included in the second way of the data array.

30. A method of operating an associative cache having a plurality of memory locations for storing data, each of the plurality of memory locations having only a single word line associated therewith, the method comprising an act of:

(A) concurrently accessing with first and second devices different ones of the plurality of memory locations of the associative cache.

31. The method of claim 30, wherein the associative cache includes at least first and second ways, and wherein the act (A) comprises an act of:

(A1) using the first and second devices to concurrently access memory locations included in the first and second ways, respectively.

32. The method of claim 30, wherein the first device includes a processor, wherein the second device includes a data transfer engine, and wherein the act (A) includes acts of:

(A1) using the processor to access the cache; and

(A2) using the data transfer engine to transfer data between the cache and a lower-level memory.

33. The method of claim 32, wherein the act (A2) includes an act of:

using a DMA controller to transfer data between the cache and the lower-level memory.

34. A method of operating an associative cache having a plurality of memory locations for storing data, and a plurality of outputs for providing data retrieved from the memory locations to respective devices, each of the plurality of memory locations having only a single word line associated therewith, the method including an act of:

(A) concurrently providing data from different ones of the plurality of memory locations to the respective devices via the plurality of outputs.

35. The method of claim 34, wherein the associative cache includes at least first and second ways, and wherein the act (A) includes an act of:

(A1) controlling first and second multiplexers to concurrently select as their respective outputs data from different ones of the at least first and second ways of the cache.

36. A method of operating an associative cache having a plurality of memory locations, each of the plurality of memory locations having only a single word line associated therewith, the method comprising an act of:

(A) using multiple decoders to decode respective addresses provided to the cache.

37. The method of claim 36, wherein each of the decoders is configured and arranged to perform decoding for one of a plurality of ways of the cache, and wherein the act (A) includes an act of:

(A1) using the multiple decoders to decode respective addresses provided to respective ones of the plurality of ways of the cache.

38 The method of claim 36, wherein the act (A) includes an act of:

(A1) using the multiple decoders to concurrently decode respective addresses provided to the cache.

39. The method of claim 36, wherein each of the multiple decoders is preceded by a multiplexer that receives first and second addresses as inputs and provides a selected one of the first and second addresses as an output to the decoder it precedes, and wherein the method further comprises an act of:

(B) controlling at least one of the multiplexers to select one of the first and second addresses as its output while concurrently controlling another of the multiplexers to select the other of the first and second addresses as its output.

40. The method of claim 39, wherein each of the decoders is configured and arranged to perform decoding for one of a plurality of ways of the cache, and wherein the act (A) includes an act of:

(A1) using the multiple decoders to decode respective addresses provided to respective ones of the plurality of ways of the cache.

41. A cache memory system, comprising:

a plurality of memory locations for storing data and addresses associated with the data, each of the plurality of memory locations having only a single word line associated therewith; and

means for enabling first and second devices to access different ones of the plurality of memory locations concurrently.

42. The system of claim 41, wherein the memory locations are configured and arranged to be included in at least first and second ways, and wherein the means for enabling includes means for enabling the first and second devices to access the first and second ways concurrently.

43. A cache memory system, comprising:

a plurality of memory locations for storing data and addresses associated with the data, the memory locations being configured and arranged to be included in at least first and second ways normally accessible by a processor;

means for selectively preventing the processor from accessing the first way while permitting the processor to access the second way; and

means, distinct from the processor, for accessing the first way while the processor is prevented from accessing the first way.

44. A method of operating an associative cache including at least first and second ways normally accessible by a processor, comprising acts of:

(A) preventing the processor from accessing the first way while permitting the processor to access the second way;

(B) while the processor is prevented from accessing the first way but is permitted to access the second way, permitting a device other than the processor to access the first way; and

(C) at a time when the step (A) is not being performed, permitting the processor to access the first way.

45. The method of claim 44, wherein the act (B) includes an act of:

(B1) using a data transfer engine to transfer data between a lower-level memory and the first way.